Day : Friday
Date: 8/10/2007

PALM INTRANET

Time: 17:15:40

Inventor Name Search Result

Your Search was:

Last Name = CHAUVEL
First Name = GERARD

Application#	Patent#	Status	Date Filed	Title	Inventor Name
09591656	6792508	150	06/09/2000	CACHE WITH MULTIPLE FILL MODES	CHAUVEL, GERARD
09677658	6452641	150	10/04/2000	METHOD AND APPARATUS FOR PROVIDING AN ON-SCREEN DISPLAY WITH VARIABLE RESOLUTION CAPABILITY	CHAUVEL, GERARD
09679000	6310657	150	10/04/2000	REAL TIME WINDOW ADDRESS CALCULATION FOR ON-SCREEN DISPLAY	CHAUVEL, GERARD
09696052	7111177	150	10/25/2000	SYSTEM AND METHOD FOR EXECUTING TASKS ACCORDING TO A SELECTED SCENARIO IN RESPONSE TO PROBABILISTIC POWER CONSUMPTION INFORMATION OF EACH SCENARIO	CHAUVEL, GERARD
09703827	Not Issued	161	11/01/2000	Prefetch for TLB cache	CHAUVEL, GERARD
09932136	7062304	150	08/17/2001	TASK BASED ADAPTATIVE PROFILIG AND DEBUGGING	CHAUVEL, GERARD
09932137	6901521	150	08/17/2001	DYNAMIC HARDWARE CONTROL FOR ENERGY MANAGEMENT SYSTEMS USING TASK ATTRIBUTES	CHAUVEL, GERARD
09932222	6681297	150	08/17/2001	SOFTWARE CONTROLLED CACHE CONFIGURATION BASED ON AVERAGE MISS RATE	CHAUVEL, GERARD
09932308	6745293	150	08/17/2001	LEVEL 2 SMARTCACHE ARCHITECTURE SUPPORTING SIMULTANEOUS MULTIPROCESSOR ACCESSES	CHAUVEL, GERARD

09932317	6678797	150	08/17/2001	CACHE/SMARTCACHE WITH INTERRUPTIBLE BLOCK PREFETCH	CHAUVEL, GERARD
09932319	6839813	150	08/17/2001	TLB OPERATIONS BASED ON SHARED BIT	CHAUVEL, GERARD
09932354	6889330	150	08/17/2001	DYNAMIC HARDWARE CONFIGURATION FOR ENERGY MANAGEMENT SYSTEMS USING TASK ATTRIBUTES	CHAUVEL, GERARD
09932359	6738864	150	08/17/2001	LEVEL 2 CACHE ARCHITECTURE FOR MULTIPROCESSOR WITH TASK_ID AND RESOURCE_ID	CHAUVEL, GERARD
09932361	7174194	150	08/17/2001	TEMPERATURE FIELD CONTROLLED SCHEDULING FOR PROCESSING SYSTEMS	CHAUVEL, GERARD
09932362	6742103	150	08/17/2001	PROCESSING SYSTEM WITH SHARED TRANSLATION LOOKASIDE BUFFER	CHAUVEL, GERARD
09932363	6766421	150	08/17/2001	FAST HARDWARE LOOPING MECHANISM FOR CACHE CLEANING AND FLUSHING OF CACHE ENTRIES CORRESPONDING TO A QUALIFIER FIELD	CHAUVEL, GERARD
09932378	6851072	150	08/17/2001	FAULT MANAGEMENT AND RECOVERY BASED ON TASK-ID	CHAUVEL, GERARD
09932380	6684280	150	08/17/2001	TASK BASED PRIORITY ARBITRATION	CHAUVEL, GERARD
09932382	6957315	150	08/17/2001	TLB LOCK AND UNLOCK OPERATION	CHAUVEL, GERARD
09932397	6779085	150	08/17/2001	TLB OPERATION BASED ON TASK-ID	CHAUVEL, GERARD
09932556	Not Issued	95	08/17/2001	ADDRESS SPACE PRIORITY ARBITRATION	CHAUVEL, GERARD
09932600	Not Issued	161	08/17/2001	Multilevel cache architecture and data transfer	CHAUVEL, GERARD
09932607	6742104	150	08/17/2001	MASTER/SLAVE PROCESSING SYSTEM WITH SHARED TRANSLATION LOOKASIDE BUFFER	CHAUVEL, GERARD
09932611	6738888	150	08/17/2001	TLB WITH RESOURCE ID	CHAUVEL,

				FIELD	GERARD
09932634	6728838	150	08/17/2001	CACHE OPERATION BASED ON RANGE OF ADDRESSES	CHAUVEL, GERARD
09932643	6754781	150	08/17/2001	CACHE WITH DMA AND DIRTY BITS	CHAUVEL, GERARD
09932650	6697916	150	08/17/2001	CACHE WITH BLOCK PREFETCH AND DMA	CHAUVEL, GERARD
09932651	6751706	150	08/17/2001	MULTIPLE MICROPROCESSORS WITH A SHARED CACHE	CHAUVEL, GERARD
09932794	6789172	150	08/17/2001	CACHE AND DMA WITH A GLOBAL VALID BIT	CHAUVEL, GERARD
09932807	6760829	150	08/17/2001	MMU DESCRIPTOR HAVING BIG/LITTLE ENDIAN BIT TO CONTROL THE TRANSFER DATA BETWEEN DEVICES	CHAUVEL, GERARD
09932866	7120715	150	08/17/2001	PRIORITY ARBITRATION BASED ON CURRENT TASK AND MMU	CHAUVEL, GERARD
10003570	Not Issued	83	10/24/2001	Data processing apparatus, system and method	CHAUVEL, GERARD
10151282	Not Issued	71	05/20/2002	Energy-aware scheduling of application execution	CHAUVEL, GERARD
10157530	7146613	150	05/29/2002	JAVA DSP ACCELERATION BY BYTE-CODE OPTIMIZATION	CHAUVEL, GERARD
10157555	6769052	150	05/29/2002	CACHE WITH SELECTIVE WRITE ALLOCATION	CHAUVEL, GERARD
10157576	6772326	150	05/29/2002	INTERRUPTIBLE AND RE-ENTRANT CACHE CLEAN RANGE INSTRUCTION	CHAUVEL, GERARD
10157773	Not Issued	41	05/29/2002	Data transfer controlled by task attributes	CHAUVEL, GERARD
10166160	6934820	150	06/10/2002	TRAFFIC CONTROLLER USING PRIORITY AND BURST CONTROL FOR REDUCING ACCESS LATENCY	CHAUVEL, GERARD
10195268	Not Issued	61	07/15/2002	Application execution profiling in conjunction with a virtual machine	CHAUVEL, GERARD
10404854	Not Issued	93	04/01/2003	TRANSPORT PACKET PARSER	CHAUVEL, GERARD
10631120	Not Issued	83	07/31/2003	Inter-processor control	CHAUVEL, GERARD
10631185	Not	161	07/31/2003	Write back policy for memory	CHAUVEL,

	Issued				GERARD
10631195	Not Issued	41	07/31/2003	Conditional garbage based on monitoring to improve real time performance	CHAUVEL, GERARD
10631196	Not Issued	41	07/31/2003	Saturated arithmetic in a processing unit	CHAUVEL, GERARD
10631205	Not Issued	71	07/31/2003	Methods and apparatuses for managing memory	CHAUVEL, GERARD
10631246	Not Issued	41	07/31/2003	Processor that accommodates multiple instruction sets and multiple decode modes	CHAUVEL, GERARD
10631252	Not Issued	161	07/31/2003	Methods and apparatuses for managing memory	CHAUVEL, GERARD
10631308	Not Issued	61	07/31/2003	Mixed stack-based RISC processor	CHAUVEL, GERARD
10631422	7162586	150	07/31/2003	SYNCHRONIZING STACK STORAGE	CHAUVEL, GERARD
10631939	Not Issued	71	07/31/2003	Multi-processor computing system having a JAVA stack machine and a RISC-based processor	CHAUVEL, GERARD

JHS

JHS

Search and Display More Records.

Search Another: Inventor **Last Name** CHAUVEL **First Name** GERARD [Search]

To go back use Back button on your browser toolbar.

Back to. PALM | ASSIGNMENT | OASIS | Home page

Day : Friday
Date: 8/10/2007

PALM INTRANET

Time: 17:16:24

Inventor Name Search Result

Your Search was:

Last Name = CHAUVEL
First Name = GERARD

Application#	Patent#	Status	Date Filed	Title	Inventor Name
10632024	Not Issued	71	07/31/2003	Synchronization of processor states	CHAUVEL, GERARD
10632067	7203797	150	07/31/2003	MEMORY MANAGEMENT OF LOCAL VARIABLES	CHAUVEL, GERARD
10632069	Not Issued	61	07/31/2003	Using IMPDEP2 for system commands related to Java accelator hardware	CHAUVEL, GERARD
10632076	Not Issued	61	07/31/2003	Memory management of local variables upon a change of context	CHAUVEL, GERARD
10632077	Not Issued	71	07/31/2003	Concurrent task execution in a multi-processor, single operating system environment	CHAUVEL, GERARD
10632079	7058765	150	07/31/2003	PROCESSOR WITH A SPLIT STACK	CHAUVEL, GERARD
10632084	Not Issued	161	07/31/2003	Test and skip processor instruction having at least one register operand	CHAUVEL, GERARD
10632214	Not Issued	61	07/31/2003	Test with immediate and skip processor instruction	CHAUVEL, GERARD
10632215	Not Issued	41	07/31/2003	Reformat logic to translate between a virtual address and a compressed physical address	CHAUVEL, GERARD
10632216	Not Issued	71	07/31/2003	Micro-sequence execution in a processor	CHAUVEL, GERARD
10632222	Not Issued	61	07/31/2003	Program counter adjustment based on the detection of an instruction prefix	CHAUVEL, GERARD
10632228	7069415	150	07/31/2003	SYSTEM AND METHOD TO AUTOMATICALLY STACK AND UNSTACK JAVA LOCAL VARIABLES	CHAUVEL, GERARD

10632229	6996683	150	07/31/2003	CACHE COHERENCY IN A MULTI-PROCESSOR SYSTEM	CHAUVEL, GERARD
10818584	Not Issued	71	04/05/2004	Management of stack-based memory usage in a processor	CHAUVEL, GERARD
10830917	Not Issued	41	04/22/2004	Dynamically changing the semantic of an instruction	CHAUVEL, GERARD
10831387	Not Issued	41	04/22/2004	Embedded garbage collection	CHAUVEL, GERARD
10831388	Not Issued	161	04/22/2004	Memory allocation in a multi-processor system	CHAUVEL, GERARD
10831470	Not Issued	123	04/23/2004	Unresolved instruction resolution	CHAUVEL, GERARD
10831575	Not Issued	30	04/22/2004	Accessing device driver memory in programming language representation	CHAUVEL, GERARD
10891821	Not Issued	41	07/14/2004	Smart cache	CHAUVEL, GERARD
11116522	Not Issued	61	04/28/2005	Compare instruction	CHAUVEL, GERARD
11116893	Not Issued	161	04/28/2005	Memory access instruction with optional error check	CHAUVEL, GERARD
11116897	Not Issued	61	04/28/2005	Pack instruction	CHAUVEL, GERARD
11116918	Not Issued	61	04/28/2005	Unpack instruction	CHAUVEL, GERARD
11135796	Not Issued	41	05/24/2005	Method and system of informing a micro-sequence of operand width	CHAUVEL, GERARD
11186036	Not Issued	30	07/21/2005	Removing local RAM size limitations when executing software code	CHAUVEL, GERARD
11186239	Not Issued	30	07/21/2005	Splitting execution of instructions between hardware and software	CHAUVEL, GERARD
11186271	Not Issued	30	07/21/2005	Method and system for accessing indirect memories	CHAUVEL, GERARD
11186315	Not Issued	30	07/21/2005	Storing contexts for thread switching	CHAUVEL, GERARD
11186330	Not Issued	30	07/21/2005	Optimizing data manipulation in media processing applications	CHAUVEL, GERARD
11188309	Not Issued	30	07/25/2005	Delegating tasks between multiple processor cores	CHAUVEL, GERARD
11188310	Not Issued	25	07/25/2005	Method and system to emulate an M-bit instruction set	CHAUVEL, GERARD

11188311	Not Issued	30	07/25/2005	Automatic operand load, modify and store	CHAUVEL, GERARD
11188336	Not Issued	41	07/25/2005	Method and system to disable the "wide" prefix	CHAUVEL, GERARD
11188502	Not Issued	30	07/25/2005	Method and system to construct a data-flow analyzer for a bytecode verfier	CHAUVEL, GERARD
11188503	Not Issued	30	07/25/2005	Method and system of using a "WIDE" opcode other than prefix	CHAUVEL, GERARD
11188504	Not Issued	30	07/25/2005	Identifying code for compilation	CHAUVEL, GERARD
11188592	Not Issued	41	07/25/2005	Compare instruction	CHAUVEL, GERARD
11188667	Not Issued	30	07/25/2005	Emulating a direct memory access controller	CHAUVEL, GERARD
11188668	7260682	150	07/25/2005	CACHE MEMORY USABLE AS SCRATCH PAD STORAGE	CHAUVEL, GERARD
11188827	Not Issued	30	07/25/2005	Automatic operand load and store	CHAUVEL, GERARD
11188923	Not Issued	71	07/25/2005	Interrupt management in dual core processors	CHAUVEL, GERARD
11189367	Not Issued	25	07/26/2005	Method and system of control flow graph construction	CHAUVEL, GERARD
11189422	Not Issued	25	07/26/2005	Method and system for implementing an interrupt handler	CHAUVEL, GERARD
11560883	Not Issued	20	11/17/2006	Method And System Of Accessing Display Window Memory	CHAUVEL, GERARD
11677367	Not Issued	25	02/21/2007	Micro-Sequence Based Security Model	CHAUVEL, GERARD
11741237	Not Issued	17	04/27/2007	METHOD AND SYSTEM FOR PERFORMING A JAVA INTERRUPT	CHAUVEL, GERARD
60029923	Not Issued	159	11/01/1996	DEVICE FOR IDENTIFYING PACKETS OF DIGITAL DATA AND A RECEIVER FOR DIGITAL TELEVISION SIGNALS EQUIPPED WITH SUCH A DEVICE	CHAUVEL, GERARD
60030104	Not Issued	159	11/01/1996	SYSTEM TO MULTIPLEX AND BLEND GRAPHIC OSD AND MOTION VIDEO PICTURES FOR DIGITAL TELEVISION	CHAUVEL, GERARD

60030105	Not Issued	159	11/01/1996	ON SCREEN DISPLAY SYSTEM WITH REAL TIME WINDOW ADDRESS	CHAUVEL, GERARD

Search and Display More Records.

	Last Name	First Name	
Search Another: Inventor	CHAUVEL	GERARD	Search

To go back use Back button on your browser toolbar.

Back to PALM | ASSIGNMENT | OASIS | Home page

PORTAL

USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+trap +(hold +or +delay) +processor SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Published before July 2002
Terms used: **trap hold** or **delay processor**

Found **436** of **132,804**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 The multics system: an examination of its structure
Elliott I. Organick
January 1972 Book
Publisher: MIT Press
Additional Information: full citation, abstract, references, cited by, index terms

This volume provides an overview of the Multics system developed at M.I.T.--a time-shared, general purpose utility like system with third-generation software. The advantage that this new system has over its predecessors lies in its expanded capacity to manipulate and file information on several levels and to police and control access to data in its various files. On the invitation of M.I.T.'s Project MAC, Elliott Organick developed over a period of years an explanation of the workings, concep ...

2 Tolerating late memory traps in ILP processors
Xiaogang Qiu, Michel Dubois
May 1999 **ACM SIGARCH Computer Archi tecture News , Proceedings of the 26th annual international symposium on Computer architecture ISCA '99**, Volume 27 Issue 2
Publisher: IEEE Computer Society, ACM Press
Full text available: pdf(100.18 KB) Publisher Site
Additional Information: full citation, abstract, references, citings, index terms

ILP processors can execute a large number of instructions at the same time. Thus it becomes more and more difficult to support traps efficiently. On the other hand a current trend in architecture is to support various memory functions in software rather than hardware, usually by trapping the execution processor on a cache miss, TLB miss or a failed access to a local or remote memory. These late memory traps block the faulting instruction at the top of the active list, backing up the pipeline. Mo ...

3 The Clipper processor: instruction set architecture and implementation
W. Hollingsworth, H. Sachs, A. J. Smith
February 1989 **Communications of the ACM**, Volume 32 Issue 2
Publisher: ACM Press
Full text available: pdf(4.67 MB)
Additional Information: full citation, abstract, references, citings, index terms, review

Intergraph's CLIPPER microprocessor is a high performance, three chip module that implements a new instruction set architecture designed for convenient programmability, broad functionality, and easy future expansion.

4 MASA: a multithreaded processor architecture for parallel symbolic computing
R. H. Halstead, T. Fujita
May 1988 **ACM SIGARCH Computer Architecture News , Proceedings of the 15th Annual International Symposium on Computer architecture ISCA '88**, Volume 16 Issue 2
Publisher: IEEE Computer Society Press, ACM Press
Full text available: pdf(1.10 MB) Additional Information: full citation, abstract, citings, index terms

MASA is a "first cut" at a processor architecture intended as a building block for a multiprocessor that can execute parallel Lisp programs efficiently. MASA features a tagged architecture, multiple contexts, fast trap handling, and a synchronization bit in every memory word. MASA's principal novelty is its use of multiple contexts both to support multithreaded execution—interleaved execution from separate instruction streams—and to speed up procedur ...

5 The influence of random delays on parallel execution times
Vikram S. Adve, Mary K. Vernon
June 1993 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1993 ACM SIGMETRICS conference on Measurement and modeling of computer systems SIGMETRICS '93**, Volume 21 Issue 1
Publisher: ACM Press
Full text available: pdf(1.52 MB) Additional Information: full citation, abstract, references, citings, index terms

Stochastic models are widely used for the performance evaluation of parallel programs and systems. The stochastic assumptions in such models exe intended to represent non-deterministic processing requirements as well as random delays due to inter-process communication end resource contention. In this paper, we provide compelling analytical and experimental evidence that in current and foreseeable shared-memory programs, communication delays introduce negligible variance into the execution time b ...

6 Selected writings on computing: a personal perspective
Edsger W. Dijkstra
January 1982 Book
Publisher: Springer-Verlag New York, Inc.
Additional Information: full citation, abstract, references, cited by, index terms

Since the summer of 1973, when I became a Burroughs Research Fellow, my life has been very different from what it had been before. The daily routine changed: instead of going to the University each day, where I used to spend most of my time in the company of others, I now went there only one day a week and was most of the time that is, when not travelling!-- alone in my study. In my solitude, mail and the written word in general became more and more important. The circumstance that my employe ...

7 A processor architecture for horizon
M. R. Thistle, B. J. Smith
November 1988 **Proceedings of the 1988 ACM/IEEE conference on Supercomputing Supercomputing '88**
Publisher: IEEE Computer Society Press
Full text available: pdf(970.44 KB) Additional Information: full citation, abstract, references, citings, index terms

Horizon is a scalable shared-memory Multiple Instruction stream - Multiple Data stream (MIMD) computer architecture independently under study at the Supercomputing

Research Center (SRC) and Tera Computer Company. It is composed of a few hundred identical scalar processors and a comparable number of memories, sparsely embedded in a three-dimensional nearest-neighbor network. Each processor has a horizontal instruction set that can issue up to three floating point operations per cycle without ...

8 APRIL: a processor architecture for multiprocessing
Anant Agarwal, Beng-Hong Lim, David Kranz, John Kubiatowicz
May 1990 **ACM SIGARCH Computer Archi tecture News , Proceedings of the 17th annual international symposium on Computer Architecture ISCA '90**, Volume 18 Issue 3a
Publisher: ACM Press

Full text available: pdf(1.38 MB) Additional Information: full citation, abstract, references, citings, index terms

Processors in large-scale multiprocessors must be able to tolerate large communication latencies and synchronization delays. This paper describes the architecture of a rapid-context-switching processor called APRIL with support for fine-grain threads and synchronization. APRIL achieves high single-thread performance and supports virtual dynamic threads. A commercial RISC-based implementation of APRIL and a run-time software system that can switch contexts in about 10 cycles is described. Me ...

9 Informing memory operations: providing memory performance feedback in modern processors
Mark Horowitz, Margaret Martonosi, Todd C. Mowry, Michael D. Smith
May 1996 **ACM SIGARCH Computer Archi tecture News , Proceedings of the 23rd annual international symposium on Computer architecture ISCA '96**, Volume 24 Issue 2
Publisher: ACM Press

Full text available: pdf(1.55 MB) Additional Information: full citation, abstract, references, citings, index terms

Memory latency is an important bottleneck in system performance that cannot be adequately solved by hardware alone. Several promising software techniques have been shown to address this problem successfully in specific situations. However, the generality of these software approaches has been limited because current architectures do not provide a fine-grained, low-overhead mechanism for observing and reacting to memory behavior directly. To fill this need, we propose a new class of memory operati ...

10 Operating system principles
Per Brinch Hansen
January 1973 Book
Publisher: Prentice-Hall, Inc.

Full text available: pdf(16.81 MB) Additional Information: full citation, abstract, references, cited by, index terms

From the Preface

MAIN GOAL

This book tries to give students of computer science and professional programmers a general understanding of *operating systems*--the programs that enable people to share computers efficiently.

To make the sharing of a computer tolerable, an operating system must enforce certain rules of behavior on all its users. One would therefore expect the designers of operating systems to do their utmost to make them as s ...

11 Smalltalk-80: the language and its implementation
Adele Goldberg, David Robson
January 1983 Book
Publisher: Addison-Wesley Longman Publishing Co., Inc.
Full text available: pdf(33.56 MB) Additional Information: full citation, abstract, cited by, index terms, review

From the Preface (See Front Matter for full Preface)

Advances in the design and production of computer hardware have brought many more people into direct contact with computers. Similar advances in the design and production of computer software are required in order that this increased contact be as rewarding as possible. The Smalltalk-80 system is a result of a decade of research into creating computer software that is appropriate for producing highly functional and interactive ...

12 A RISC processor architecture with a versatile stack system
Claus Aßmann
December 1993 **ACM SIGARCH Computer Architecture News**, Volume 21 Issue 5
Publisher: ACM Press
Full text available: pdf(785.39 KB) Additional Information: full citation, abstract, citings, index terms

Modern RISC processors mainly differ w.r.t, the organization of their register files. There are currently three different approaches: a flat register file (e.g., MIPS), fixed-size register windows (SPARC), or a stack-like organization (AM29K). This paper describes a processor architecture with a new stack system, which is tailored to the needs of executing functional languages. In order to support a fast subroutine call mechanism and efficient parameter passing, our architecture uses a system of ...

Keywords: RISC, functional languages, processor architecture, register file, stack

13 Using the Alfa-1 simulated processor for educational purposes
Gabriel A. Wainer, Sergio Daicz, Luis F. De Simoni, Demian Wassermann
December 2001 **Journal on Educational Resources in Computing (JERIC)**, Volume 1 Issue 4
Publisher: ACM Press
Full text available: pdf(238.65 KB) Additional Information: full citation, abstract, references, index terms

Alfa-1 is a simulated computer designed for computer organization courses. Alfa-1 and its accompanying toolkit allow students to acquire practical insights into developing hardware by extending existing components. The DEVS formalism is used to model individual components and to integrate them into a hierarchy that describes the detailed behavior of different levels of a computer's architecture. We introduce Alfa-1 and the toolkit, show how to extend existing components, and describe how ...

Keywords: DEVS formalism, modeling computer architectures, systems specification

14 Essays in computing science
C. A. R. Hoare
January 1989 Book
Publisher: Prentice-Hall, Inc.
Full text available: pdf(20.91 MB) Additional Information: full citation, abstract, references, cited by, review

Charles Antony Richard Hoare is one of the most productive and prolific computer scientists. This volume contains a selection of his published papers. There is a need, as in a Shakespearian Chorus, to offer some apology for what the book manifestly fails to achieve. It is not a complete 'collected works'. Selection between papers of this quality is not easy and, given the book's already considerable size, some difficult decisions as to what to omit have had to be made. Pity the editor weighin ...

15 A modeling approach and design tool for pipelined central processors

D. E. Lang, T. K. Agerwala, K. M. Chandy
April 1979 **Proceedings of the 6th annual symposium on Computer architecture ISCA '79**
Publisher: ACM Press

Full text available: pdf(645.97 KB) Additional Information: full citation, abstract, references, citings, index terms

As CPUs have become larger and more complex, it has become increasingly more difficult during hardware design and implementation to predict how well a CPU will perform. Furthermore, buyers of such machines have a similar problem in evaluating CPU performance among a diverse selection of computers, since MIP rates quoted by manufacturers may be misleading. This paper describes a methodology based on simulation models for predicting the performance of central processors with instru ...

16 Trace-driven memory simulation: a survey

Richard A. Uhlig, Trevor N. Mudge
June 1997 **ACM Computing Surveys (CSUR)**, Volume 29 Issue 2
Publisher: ACM Press

Full text available: pdf(636.11 KB) Additional Information: full citation, abstract, references, citings, index terms, review

As the gap between processor and memory speeds continues to widen, methods for evaluating memory system designs before they are implemented in hardware are becoming increasingly important. One such method, trace-driven memory simulation, has been the subject of intense interest among researchers and has, as a result, enjoyed rapid development and substantial improvements during the past decade. This article surveys and analyzes these developments by establishing criteria for evaluating trac ...

Keywords: TLBs, caches, memory management, memory simulation, trace-driven simulation

17 On randomization in sequential and distributed algorithms

Rajiv Gupta, Scott A. Smolka, Shaji Bhaskar
March 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 1
Publisher: ACM Press

Full text available: pdf(8.01 MB) Additional Information: full citation, abstract, references, citings, index terms

Probabilistic, or randomized, algorithms are fast becoming as commonplace as conventional deterministic algorithms. This survey presents five techniques that have been widely used in the design of randomized algorithms. These techniques are illustrated using 12 randomized algorithms—both sequential and distributed— that span a wide range of applications, including:primality testing (a classical problem in number theory), interactive probabilistic proof s ...

Keywords: Byzantine agreement, CSP, analysis of algorithms, computational complexity, dining philosophers problem, distributed algorithms, graph isomorphism, hashing, interactive probabilistic proof systems, leader election, message routing, nearest-neighbors problem, perfect hashing, primality testing, probabilistic techniques, randomized or probabilistic algorithms, randomized quicksort, sequential algorithms, transitive tournaments, universal hashing

18 Fast detection of communication patterns in distributed executions
Thomas Kunz, Michiel F. H. Seuren
November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research CASCON '97**
Publisher: IBM Press
Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

19 Waiting algorithms for synchronization in large-scale multiprocessors
Beng-Hong Lim, Anant Agarwal
August 1993 **ACM Transactions on Computer Systems (TOCS)**, Volume 11 Issue 3
Publisher: ACM Press
Full text available: pdf(2.72 MB) Additional Information: full citation, abstract, references, citings, index terms

Through analysis and experiments, this paper investigates two-phase waiting algorithms to minimize the cost of waiting for synchronization in large-scale multiprocessors. In a two-phase algorithm, a thread first waits by polling a synchronization variable. If the cost of polling reaches a limit Lpoll and further waiting is necessary, the thread is blocked, incurring an additional fixed cost, B. The choice of Lpoll
Keywords: barriers, blocking, competitive analysis, locks, producer-consumer synchronization, spinning, waiting time

20 Informing memory operations: memory performance feedback mechanisms and their applications
Mark Horowitz, Margaret Martonosi, Todd C. Mowry, Michael D. Smith
May 1998 **ACM Transactions on Computer Systems (TOCS)**, Volume 16 Issue 2
Publisher: ACM Press
Full text available: pdf(344.74 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Memory latency is an important bottleneck in system performance that cannot be adequately solved by hardware alone. Several promising software techniques have been shown to address this problem successfully in specific situations. However, the generality of these software approaches has been limited because current architecturtes do not provide a fine-grained, low-overhead mechanism for observing and reacting to memory behavior directly. To fill this need, this article proposes a new class ...

Keywords: cache miss notification, memory latency, processor architecture

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

IEEE Xplore® RELEASE 2.4

Welcome United States Patent and Trademark Office

Edit an existing query or compose a new query in the Search Query Display.

Select a search number (#) to:

- Add a query to the Search Query Display
- Combine search queries using AND, OR, or NOT
- Delete a search
- Run a search

Fri, 10 Aug 2007, 5:36:30 PM EST

Search Query Display

Recent Search Queries

#1	(((trap <and> processor <and> (hold <or> delay))<in>metadata)) <and> (pyr >= 1970 <and> pyr <= 2002)
#2	((trap<in>metadata) <and> (processor<in>metadata))
#3	(((trap<in>metadata) <and> (processor<in>metadata))) <and> (pyr >= 1970 <and> pyr <= 2002)

Indexed by Inspec®